

IN THE CLAIMS:

Claims 1, 4, 5, 6, 11 - 13, 15, 15, 19 - 21, and 23 - 30 have been amended.

1. (currently amended) A method of driving a motor ~~emulating advancement of a speed sensor~~, comprising:

calculating an emulated advancement time based on the motor's efficiency;

~~measuring a motor speed utilizing a tachometer signal transmitted from a speed sensor~~ measuring a time for one revolution of a rotor of the motor;

subtracting the emulated advancement time from the ~~motor speed~~ time for the one revolution of the rotor to generate a commutation countdown time; and

commutating outputs from a controller to the motor when the commutation countdown time has elapsed.

2. (original) The method of claim 1, further including measuring an actual advance time, the actual advance time being a time between the commutating of the outputs and a receipt by the controller of a next speed sensor interrupt.

3. (original) The method of claim 2, further including calculating an anticipated motor speed by adding the actual advance time to the commutation countdown time.

4. (currently amended) The method of claim 3, further including

(a) calculating a new commutation countdown time by subtracting the emulated advancement time from the anticipated ~~[[fan]]~~ motor speed;

(b) commutating outputs from the controller when the new commutation countdown time has elapsed;

(c) measuring the actual advance time between the commutating of the outputs

and the next speed sensor interrupt;

(d) calculating the anticipated motor speed by adding the actual advance time to the new commutation countdown time; and

continuing the steps (a), (b), (c), and (d) until the anticipated motor speed is lower than a pre-determined motor speed threshold.

5. (currently amended) The method of claim 4, further including decrementing the emulated advancement time by a pre-determined advancement time to create a decremented emulated advancement time if the anticipated ~~[[fan]]~~ motor speed is lower than a pre-determined ~~fan speed~~ threshold;

calculating the new commutation countdown time by subtracting the decremented emulated advancement time from the anticipated ~~[[fan]]~~ motor speed;

commutating the outputs from the controller when the new commutation countdown time has elapsed;

measuring the actual advance time, the actual advance time being the time between the commutating of the outputs from the controller and the receipt by the controller of the next ~~[[hall]]~~ speed sensor interrupt; and

calculating the anticipated motor speed by adding the actual advance time to the new commutation countdown time.

6. (currently amended) A method of initializing neutral commutation, comprising:

initializing a first driving signal to drive a motor;

receiving a tachometer signal from a speed sensor for the motor;

~~calculating~~ measuring a ~~motor speed~~ pulse time based on the received

tachometer signal;

calculating a commutation countdown value by subtracting an initial advancing time from the ~~calculated motor speed~~ pulse time if the calculated motor speed is lower than a minimum pre-determined $[[\text{speed}]]$ threshold; and

commutating outputs to the motor including generating a second driving signal if the commutation countdown value has elapsed.

7. (original) The method of claim 6, further including measuring an actual advance time as a period between the commutating of the outputs and a receipt of a next speed sensor interrupt.

8. (original) The method of claim 7, further including creating an anticipated motor speed by adding the actual advance time to the commutation countdown time.

9. (original) The method of claim 8, further including determining if the initial advancing time is less than a threshold advancing time.

10. (original) The method of claim 9, further including

(a) creating a new advancing time by adding an incremental advancing time to the initial advancing time if the initial advancing time is less than the threshold advancing time;

(b) calculating a new commutation countdown time by subtracting the new advancing time from the anticipated motor speed;

(c) commutating the outputs including generating the first drive signal when the new commutation countdown time has elapsed;

(d) measuring the actual advance time between the commutating of the outputs and the next speed sensor interrupt;

(e) creating a new anticipated motor speed by adding the actual advance time to the new commutation countdown time, and

repeating steps (a), (b), (c), (d), and (e) until the new advancing time is greater than threshold advancing time.

11. (currently amended) A method of ~~emulating advancement of a speed sensor driving a motor~~, comprising:

calculating an advancement time based on a motor's efficiency;

utilizing a tachometer signal transmitted from a speed sensor for the motor to ~~generate a calculated motor speed~~ to measure a time between pulses of the tachometer signal;

subtracting the advancement time from the ~~calculated motor speed~~ the time between pulses to generate a commutation countdown time; and

commutating outputs from a controller to the motor when the commutation countdown time has elapsed.

12. (currently amended) The method of claim 11, further including

(a) utilizing the tachometer signal transmitted from the speed sensor at a new measurement time to ~~generate the calculated motor speed~~ measure a new time between pulses,

(b) calculating a new commutation countdown time by subtracting the advancement time from the ~~calculated motor speed~~ new time between pulses;

(c) commutating outputs to the motor from the controller when the new commutation countdown time has elapsed; and

continuing the steps (a), (b), and (c) until the ~~calculated motor speed~~ new time

between pulses is lower than a pre-determined ~~motor-speed~~ threshold.

13. (currently amended) A method of initializing neutral commutation, comprising:

- initializing a first driving signal to drive a motor;
- receiving a tachometer signal from a speed sensor for the motor;
- calculating ~~a motor-speed~~ a time between pulses of a tachometer signal based on the received tachometer signal;
- calculating a commutation countdown value by subtracting an initial advancing time from the ~~calculated motor-speed~~ time between pulses of the tachometer signal if the ~~calculated motor-speed~~ time between pulses of the tachometer signal is lower than a minimum pre-determined ~~[[speed]]~~ threshold; and
- commutating outputs to the motor, including generating a second driving signal, if the commutation countdown value has elapsed.

14. (original) The method of claim 13, further including determining if the initial advancing time is less than a threshold advancing time.

15. (currently amended) The method of claim 14, further including
- (a) creating a new advancing time by adding an incremental advancing time to the initial advancing time if the initial advancing time is less than the threshold advancing time;
 - (b) calculating a new commutation countdown time by subtracting the new advancing time from the ~~calculated motor-speed~~ time between pulses in the tachometer signal;
 - (c) commutating the outputs to the motor, including generating the first drive

signal when the new commutation countdown time has elapsed, and

repeating steps (a), (b), and (c) until the new advancing time is greater than threshold advancing time.

16. (currently amended) A microcontroller to drive a motor, comprising:

a ~~[[speed]]~~ determination module to receive a tachometer signal from a speed sensor for the motor, to ~~generate a calculated motor speed~~ to determine a time between pulses of the tachometer signal from the tachometer signal, and to transmit the ~~calculated motor speed~~ time between pulses of the tachometer signal;

an advancing analyzation module to receive the ~~calculated motor speed~~ time between pulses of the tachometer signal, to calculate a commutation countdown time by subtracting an advancement time from the ~~calculated motor speed~~ time between pulses of the tachometer signal, and to transmit the commutation countdown time,

a counting module to receive the commutation countdown time, and to transmit a commutation signal once the commutation countdown time has expired; and

a commutation output module to receive the commutation signal and to switch outputs of the microcontroller to the motor upon receipt of the commutation signal.

17. (original) The microcontroller of claim 16, wherein the commutation output module notifies the advancing analyzation module that the outputs to the motor are switched, the advancing analyzation module transmits a second signal to the counting module to begin an actual advance count, the speed determination module transmits a signal identifying that a next speed sensor interrupt has been received, and the counting module stops the actual advance count upon receipt of the next speed sensor interrupt.

18. (currently amended) The microcontroller of claim 17, wherein an anticipated motor speed is calculated by adding the commutation countdown time and the ~~anticipated motor speed~~ actual advance count.

19. (currently amended) The microcontroller of claim ~~[[16]]~~ 17, further including, after the commutation of the microcontroller outputs, the ~~[[speed]]~~ determination module determining the ~~calculated motor speed~~ time between the pulses of the tachometer signal for a new measurement time, and transmitting the ~~calculated motor speed~~ time between the pulses of the tachometer signal for the new measurement time to the advancing analyzation module; and

the advancing analyzation module calculating the commutation countdown time for the new measurement time by subtracting the advance time from the ~~calculated motor speed~~ time between the pulses of the tachometer signal.

20. (currently amended) The microcontroller of claim 19, wherein the ~~[[speed]]~~ determination module utilizes a second counting module to determine the ~~calculated motor speed~~ time between the pulses of the tachometer signal for the new measurement time.

21. (currently amended) A device ~~driven by a motor~~, comprising:
a driving device to receive a driving signal and to transmit the driving signal;
a motor to receive the driving signal and to operate the device based on the driving signal;
a speed sensor to monitor the speed of the motor and to transmit a tachometer signal; and
a microcontroller to receive the tachometer signal and to generate the driving

signal, including,

a speed determination module to receive the tachometer signal from the speed sensor, to ~~generate a calculated device speed from~~ measure a time between pulses of the tachometer signal, and to transmit the ~~calculated motor speed~~ time between the pulses of the tachometer signal;

an advancing analyzation module to receive the ~~calculated motor speed~~ time between the pulses of the tachometer signal, to calculate a commutation countdown time by subtracting an advancement time from the ~~calculated motor speed~~ time between the pulses of the tachometer signal, and to transmit the commutation countdown time,

a counting module to receive the commutation countdown time, and to transmit a commutation signal once the commutation countdown time has expired; and

a commutation output module to receive the commutation signal and to switch outputs of the driving signal generated by the microcontroller upon receipt of the commutation signal.

22. (original) The device of claim 21, wherein the commutation output module notifies the advancing analyzation module that the outputs are switched, the advancing analyzation module transmits a second signal to the counting module to begin an actual advance count, the speed determination module transmits a signal identifying that a next speed sensor interrupt has been received, and the counting module stops the actual advance count upon receipt of the next speed sensor interrupt.

23. (currently amended) The device of claim 22, wherein an anticipated motor speed is calculated by adding the commutation countdown time and the ~~anticipated~~

~~motor speed~~ actual advance count.

24. (currently amended) The device of claim 21, further including, after commutation of the microcontroller outputs, the ~~[[speed]]~~ determination module determining the ~~calculated motor speed~~ time between the pulses of the tachometer signal for a new measurement time, and transmitting the ~~calculated motor speed~~ time between the pulses of the tachometer signal for the new measurement time to the advancing analyzation module, and

the advancing analyzation module calculating the commutation countdown time for the new measurement time by subtracting the advance time from the ~~calculated motor speed~~ time between the pulses of the tachometer signal.

25. (currently amended) A computer-readable medium having encoded thereon a computer-readable program code which when executed causes a microcontroller to:

calculate an emulated advancement time based on a motor's efficiency;

~~measure a motor speed utilizing a tachometer signal transmitted from a speed sensor for~~ measure a time for one revolution of a rotor of the motor;

subtract the emulated advancement time from the ~~motor speed~~ time for the one revolution of the rotor to generate a commutation countdown time; and

commutate outputs from a controller when the commutation countdown time has elapsed.

26. (currently amended) The computer readable medium of claim 25, further having encoded thereon computer readable program code, which when executed causes the microcontroller to

measure an actual advance time, the actual advance time being a time between

the commutating of the outputs and a receipt by the controller of a next speed sensor interrupt.

27. (currently amended) The computer readable medium of claim 26, ~~further~~ having encoded thereon computer readable program code, which when executed causes the microcontroller to calculate an anticipated motor speed by adding the actual interrupt time to the commutation countdown time.

28. (currently amended) A computer-readable medium having encoded thereon computer-readable program code which when executed causes a microcontroller to:

initialize a first driving signal to drive a motor;

receive a tachometer signal from a speed sensor for the motor;

calculate a ~~motor speed~~ time between pulses of the tachometer signal based on the received tachometer signal;

calculate a commutation countdown value by subtracting an initial advancing time from the ~~calculated motor speed~~ time between the pulses of the tachometer signal if the ~~calculated motor speed~~ time between the pulses of the tachometer signal is lower than a minimum pre-determined $[[\text{speed}]]$ threshold; and

commutate outputs for the motor, including generating a second driving signal if the commutation countdown value has elapsed.

29. (currently amended) The computer-readable medium of claim 28 ~~further~~ having encoded thereon computer-readable program code, which when executed causes the microcontroller to measure an actual advance time as a period between the commutating of the outputs and a receipt of a next speed sensor interrupt.

30. (currently amended) The computer-readable medium of claim 29 ~~further~~

having encoded thereon computer readable program code, which when executed causes the microcontroller to create an anticipated motor speed by adding the actual advance time to the commutation countdown time.